

Taller de Estructuras

Deibi Doria

12



1. Estructuras de Control

Es la que permite modificar el flujo de ejecución de las instrucciones. En un algoritmo se le conoce como orden en el cual se efectúan todas las acciones primitivas. Las estructuras de control permiten mostrar secuencias de órdenes a ejecutar.

2. Estructura Repetitiva

Es un ciclo donde entran las variables a ser analizadas se hacen las operaciones correspondientes y se muestran sus respectivos resultados. Ej.

```
INICIO
  REPITA
    IMPRIMIR "Digite color de la etiqueta y el valor de la compra: "
    LEER CETI, VALC
    SI (CETI ="rojo") ENTONCES
      DCTO=VALC*0.2
    SINO
      SI (CETI ="azul") ENTONCES
        DCTO=VALC*0.1
      SINO
        DCTO=VALC*1
    FIN-SI
  FIN-SI
  TOTPAG=VALC-DCTO
  IMPRIMIR "Total a pagar: ", TOTPAG
  REPITA HASTA (CETI = "Verde")
FIN
```

3. Esquemas Cualitativos

Es utilizado cuando se conoce el número de veces que debe repetirse la secuencia de un determinado ciclo. Ej.

```

INICIO
  SUMA, N = 0
  LEA: NUM
  MIENTRAS NUM > 0 HAGA
    SUMA = SUMA + NUM
    N = N + 1
    LEA: NUM
  FIN_MIENTRAS
  PROM = SUMA / N
  ESCRIBA: "LA SUMA ES:", SUMA "Y EL PROMEDIO", PROM
FIN_INICIO

```

4. Esquemas Cuantitativos

Es utilizado cuando se conoce el número de veces que debe repetirse un ciclo determinado. Ej.

```

INICIO
  SUMA = 0
  NUM = 1
  LEA: N
  MIENTRAS NUM ≤ N HAGA
    SUMA = SUMA + NUM
    NUM = NUM + 1
  FIN_MIENTRAS
  ESCRIBA: "LA SUMA ES:", SUMA
FIN_INICIO

```

5. Estructura Mientras

Es la estructura algorítmica que se ejecuta cuando la condición evaluada resulte verdadera. Ej.

```

Inicio
  Suma  $\leftarrow$  0
  I  $\leftarrow$  1
  Mientras I  $\leq$  100 hacer
    Suma  $\leftarrow$  suma + I * I
    I  $\leftarrow$  I + 1
  Fin_mientras
  Escribir (suma)
Fin

```

6. Rompimiento de Ciclos

Se hace cambiando el sentido de la expresión lógica que controla el ciclo para que esta sea falsa y no se continúe ejecutando la secuencia de instrucciones. Ej.

```

INICIO
  LEA: NN
  CONTA = 1
  ENCONTRADO = .F.
  MIENTRAS (CONTA <= NN) ^ (ENCONTRADO = .F.) HAGA
    LEA: NUM
    SI NUM = 0 ENTONCES
      ENCONTRADO = .V.
    SINO
      CONTA = CONTA + 1
    FIN_SI
  FIN_MIENTRAS
  SI ENCONTRADO = .V. ENTONCES
    ESCRIBA: "SI HAY UN CERO ENTRE LOS NUMEROS"
  SINO
    ESCRIBA: "NO HAY UN CERO ENTRE LOS NUMEROS"
  FIN_SI
FIN INICIO

```

7. Rompimiento de control de ejecución

Se presentan cuando es necesario detener el proceso porque se ha producido un cambio y se deben dar reportes de lo que ha acontecido antes de la ruptura. Ej.

```
INICIO
  LEA: COD, CODS, NOM, SAL
  MIENTRAS COD <> 0 HAGA
    ESCRIBA: "REPORTE DE SALARIOS SUCURSAL:",
    CODS
    ESCRIBA: "CODIGO NOMBRES SALARIO"
    CODSA = CODS
    TOTSUC = 0
    MIENTRAS (CODS = CODSA) ^ (COD <> 0) HAGA
      ESCRIBA: COD, NOM, SAL
      TOTSUC = TOTSUC + SAL
      LEA: COD
      SI COD <> 0 ENTONCES
        LEA: CODS, NOM, SAL
      FIN_SI
    FIN_MIENTRAS
    ESCRIBA: "TOTAL SALARIOS DE LA SUCURSAL:$",
    TOTSUC
  FIN_MIENTRAS
FIN_INICIO
```

8. Estructura Hacer Mientras

Evalúa la condición al final del lazo y no al principio. Ej.

```
Escribir("Para salir escriba un  
número negativo")
```

```
suma = 0.0
```

```
Leer(x)
```

```
Hacer
```

```
    Si (x > 0.0)
```

```
        suma = suma + x
```

```
        Leer(x)
```

```
    Fin_Si
```

```
Mientras (x > 0.0)
```